## AMENDMENTS TO THE DRAWINGS

As a preliminary matter, Applicants acknowledge that the Examiner requested that FIGS. 1, 2A and 2B be labeled "Prior Art." Replacement drawing sheets for FIGS. 1, 2A and 2B are attached after the last page of this response.

## REMARKS

This amendment is responsive to the Office Action dated March 23, 2006. Claims 1, 2, 4–9 and 12–20 are pending.

### Claim Rejection Under 35 U.S.C. § 103

In the Office Action, the Examiner rejected claims 1, 2, 4, 5, 7–9, 12, 13, 16–19 under 35 U.S.C. 103(a) as being unpatentable over Jardin (USPN 6,681,327) in view of Lockhart et al. (USPN 5,841,873). The Examiner rejected claims 6, 14, 15, and 20 under 35 U.S.C. 103(a) as being unpatentable over Jardin in view of Lockhart and in further view of Lin et al. (USPN 6,052,785).

Applicants respectfully traverse the rejection. To establish a prima facie case of obviousness, the combination of references must teach or suggest **all** the claim limitations.[1] In this case, Jardin in view of Lockhart and Lin fails disclose or suggest many elements defined by Applicants' claims. Moreover, Jardin in view of Lockhart and Lin fails to provide any teaching that would have suggested the desirability of modification to arrive at the claimed invention.

Applicants disagree with many of the Examiner's assertions regarding Jardin with respect to Applicant's independent claims. Applicants' independent claim 1 is directed to a method in which an intermediary device negotiates a secure communications session with a client apparatus, and receives encrypted packet application data for a security record that has a length greater than a packet length associated with multiple data packets. Claim 1 also requires decrypting the encrypted packet application data in each data packet with the intermediary device, and then forwarding decrypted, unauthenticated application data to the server via the secure network.

First, Jardin does not describe receiving encrypted packet application data for a security record spanning multiple packets, i.e., where the security record that has a length greater than a packet length associated with multiple data packets. Jardin does not use the term "security record" nor describe a security record or any security mechanism that spans multiple packets. Jardin describes a server broker configured to broker client transactions received over a secure network link for distribution among one or more of a plurality of fulfillment servers. In the

[1] MPEP 2142.

passage cited by the Examiner, col. 6, ll. 65–59, Jardin only states that the server broker "decrypts the packets received from the client." The fact that Jardin decrypts individual packets teaches nothing with regard to use of security records where individual security records span multiple packets. Jardin provides no teaching or discussion whatsoever of an SSL security record or other type of security record where application data for one record spans multiple packets.

Second, on a related note, the Examiner is erroneous to assert that Jardin describes an intermediary device that performs the step of forwarding decrypted, unauthenticated application data for a security record to the server. The Examiner again relies on Jardin at col. 6, ln. 65–col. 7., ln. 5, this time asserting that Jardin teaches an intermediary device (the Jardin server broker) that performs the step of forwarding decrypted *unauthenticated* application data to a server. However, at this passage, Jardin only states that server broker applies the "agreed upon ciphering algorithm" to decrypt packets and then directs the decrypted packets to the server. Nowhere does Jardin suggest that this "server broker" provides secure communications with the client (e.g., via SSL) but forwards decrypted, unauthenticated application data to the server, as required by claim 1. Applicants' claim 1 specifically requires forwarding decrypted but unauthenticated application data to the server, i.e., application data that has been decrypted by the intermediary device but not yet authenticated. Even assuming that the Jardin server broker uses a security record spanning multiple packets (which Applicants questioned above), then Jardin certainly provides no teaching or remote suggestion that the Jardin server broker decrypts packets for the security mechanism but then forwards unauthenticated application data to the server, as required by claim 1. For example, assuming the Jardin server broker uses SSL, which is listed as an example, why would the Examiner assume that the Jardin server broker decrypt SSL segments without authenticating them, as is typical? Jardin provides no teaching or even any suggestion to send decrypted unauthenticated application data sent, i.e., application data that has not yet been authenticated, e.g., for a multi-packet security record using SSL or any other mechanism. For this reason, the Examiner is erroneous to assert that Jardin describes an intermediary device that performs the step of forwarding decrypted, unauthenticated application data for a security record to the server.

In addition, Applicants disagree with the Examiner's characterization of the teachings of Lockhart. Applicants' independent claim 1 further requires discarding at least a portion of the decrypted, unauthenticated packet application data for a security record prior to receiving a final packet of the security record. Claim 1 also requires authenticating the security record on receipt of the final packet of the security record. That is, the literal requirements of claim 1 requires discarding a portion of packet application data for a security record prior to receiving the final packet for that security record then, upon receiving the final packet, authenticating the security record.

With respect to claim 1, the Examiner correctly acknowledged that Jardin fails to teach or suggest discarding at least a portion of decrypted, unauthenticated packet application data for a security record prior to receiving a final packet of that security record. However, the Examiner asserts that Lockhart teaches discarding at least a portion of the decrypted, unauthenticated packet application data for a security record prior to receiving a final packet of that security record. For support, the Examiner cites Lockhart at col. 5, ll. 33–65 without comment.

In the portion relied on by the Examiner, Lockhart describes use of a known ASCII reference value (e.g., the string "EN") to aid detection of encryption errors in individual packets. According to Lockhart, the sending device appends the predetermined ASCII reference value to each packet prior to encryption. Upon receipt, the receiving device first decrypts the entire packet (Step 215). Next, the reference value is "removed" from the decrypted packet and compared to original value. If the reference value matches the original value, then <u>the packet is processed in a normal manner</u>. If not, then an encryption error is detected and a message is sent to the sending device.

Thus, Lockhart fails to overcome the deficiencies of Jardin for numerous reasons. First, contrary to the Examiner's assertion, at no time does the Lockhart system discard anything prior to authentication, let alone discarding a portion of a security record prior to receiving a final packet of the security record. Nothing in the Lockhart reference suggests that a portion of a security record is discarded prior to authentication of the security record. For example, neither the reference value appended to each packet nor the packet itself in Lockhart is discarded prior to authenticating the security record upon receiving a last packet of the security record, as suggested by the Examiner. Lockhart makes clear that the reference value appended to each packet is

removed but then compared to the predetermined reference value to detect encryption errors. Thus, the reference value certainly is not discarded prior to authentication. To the contrary, the reference value is used for comparison purposes to detect encryption errors. Moreover, Lockhart makes clear that, if a match is detected, the "receiving device proceeds on to step (225) and continues processing the data in a normal manner." Therefore, the packet itself (i.e., the "data") is also not discarded, but rather it is used in normal manner after the comparison. So, Jardin in view of Lockhart does not teach or suggest the step of discarding any portion of a security record prior to receiving a last packet of the security record at which time the security record is then authenticated, as required by claim 1.

Second, Lockhart provides no teaching even relevant to handling a security record that spans multiple packets. Lockhart is concerned with detecting data decryption errors in <u>individual</u> data packets by appending a known reference value to each packet.[2] There simply is no concept of a single security record that spans <u>multiple</u> packets. The Examiner's statement that Lockhart teaches discarding at least a portion of the decrypted, unauthenticated packet application data for a security record prior to receiving a final packet of that security record is erroneous on its face. Lockhart does not even use the term "security record" nor discuss any form of a single security record that spans multiple packets. Lockhart makes clear that the ASCII reference value is appended <u>to each packet</u> at the sender. There simply is no concept of a security record that spans multiple packets in Lockhart, and Jardin in view of Lockhart. Contrary to the Examiner's assertion, Lockhart's use of appending ASCII values to each packet does not teach or suggest any technique for discarding at least a portion of the decrypted, unauthenticated packet application data for a security record prior to receiving a final packet of that security record, as required by claim 1.

Third, the "reference value" of Lockhart is not "decrypted, unauthenticated application data" that has been <u>already been forwarded</u> to a server, as required by claim 1. As discussed above, claim 1 requires forwarding decrypted, unauthenticated application data to the server via the secure network, and then discarding at least a portion of <u>**the decrypted, unauthenticated** packet application data</u> for the security record prior to receiving a final packet of the security record. That is, the decrypted application data is first forwarded by the intermediate device in

---

[2] Abstract.

unauthenticated form, and then a portion of that same data is discarded prior to receiving a final packet of the entire security record.

In Lockhart, the "reference value" is a predetermined 2 byte string ("EN") that is inserted by the client and removed. It is not at least a portion of the decrypted, unauthenticated application that has already been forwarded to the server without authentication, as required by claim 1. For this reason, even if Jardin were modified in view of Lockhart, as suggested by the Examiner, the result would be a system in which each client appends a reference value to each packet in a secure communication, and that reference value would presumably be used by the server broker to detect cryptographic errors in the packets. Even assuming that the Jardin server broker were to remove the reference value, it certainly would not qualify as an intermediary device that forwards decrypted, unauthenticated application data of a security record to server, discards a least a portion of that same unauthenticated application data, and then authenticates the multi-packet security record upon receiving the final packet for that security record, as required by Applicants' claim 1. Independent claims 7 and 16 are patentable over Jardin in view of Lockhart for these or similar reasons.

With respect to dependent claims 4 and 12, the Examiner asserts that Jardin teaches buffering only the remaining portion of the packet application data for the security that is a minimal length sufficient to complete a block cipher used to encrypt the data. For support, the Examiner cites col. 2, ln. 65–col. 3, ln. 3. However, this portion of Jardin only states that decrypted packets are re-directed to another computer, and that the packets may be buffered until the receiving computer is ready. This has absolutely nothing to do with block ciphers whatsoever or buffering a minimal length of application data necessary to complete a block cipher. In fact, neither Jardin nor Lockhart even suggests buffering just a remaining portion (i.e., a non-discarded portion) of unauthenticated application data that had already been forwarded by an intermediate device to a server. The Examiner admits that Jardin fails to teach discarding any portion of a security record at all, and attempts to rely on Lockhart for this teaching. This is prima facie evidence that his assertion that Jardin teaches buffering a *remaining portion* of that same application data for that security record is incorrect. As neither Jardin nor Lockhart teach or suggest discarding at least a portion of the decrypted, unauthenticated application data for a security record prior to authentication of that security record, the references clearly fail to teach

or suggest buffering only a *remaining portion of application for a security record* having a minimal length sufficient to complete a block cipher used to encrypt the data, as required by claim 4. Applicants refer the Examiner to the "small buffer" embodiment described above. None of the references teach or suggest forwarding unauthenticated application data to a server, and then only storing a partial portion of that application data for a security record prior to authentication of that security record, as required by Applicants' claims 1 and 4.

The Examiner's rejection of claim 5 and 19 is again very confusing. The Examiner only states that "[t]he Examiner holds that authenticating could only take place once the final segment was received . . . ." Thus, the Examiner appears to be taking Official Notice that the elements of claim 5 are well known. However, the Examiner did not even address the requirement of claim 5 of authenticating the decrypted data for the security record upon receiving a final TCP segment of a multi-segment encrypted data stream *and after forwarding the decrypted, unauthenticated application data received prior to the final TCP segment*. None of the references, either singularly or in combination, teach or suggest authenticating the decrypted data for the security record upon receiving a final TCP segment of a multi-segment encrypted data stream and after forwarding the decrypted, unauthenticated application data that was received prior to the final TCP segment. Claim 5 specifically requires that the decrypted data of a security record is authenticated **after** the TCP data received prior to the last TCP segment for that same record has **already been forwarded**. As explained above with respect to claim 1, none of the references, either singularly or in combination, teach or suggest an acceleration device that forwards application data of a security record to a server prior to authenticating subsequent application data of that same security record. The Official Notice taken by the Examiner appears to be irrelevant.

Applicants' claims 6 and 14 require after forwarding the decrypted, unauthenticated application data to the server, notifying the client apparatus if a failure in authenticating the security record occurs. With respect to claim 6, the Examiner asserts that Lin teaches notifying a client if a failure occurs. However, at the cited portion, Lin describes establishing an SSL session, during which user credentials are located. If the credentials are not located or incorrect, then authentication fails (blocks 418, 420 cited by the Examiner). Clearly, the Lin technique for establishing an SSL session does not teach or suggest notifying the client apparatus if a failure in

authenticating the security record occurs after forwarding the decrypted, unauthenticated application data from the intermediary to the server. There is nothing to suggest issuing an error notification after forwarding the decrypted, unauthenticated application data from an intermediate device to a server. In fact, Lin makes quite clear that the SSL session is not even established if an error occurs.

Applicants again point out that, in the context of Applicants' bufferless or small buffer embodiment, the application data is forwarded prior to authentication, thereby possibly reducing or minimizing hardware buffering requirements (see discussion of claim 1 above). This is fundamentally different from any combination of references put forth by the Examiner. Claim 6 is directed to the unique step that, in such an embodiment, an error notification is issued to the client after forwarding the data in unauthenticated form from the intermediate device to the server. Thus, even though the data has already been forwarded to the server, an error message is nevertheless issued to the client. The SSL session establishment technique of Lin referred to by the Examiner occurs at the time of authentication which, as demonstrated by Lin, is prior to even allowing SSL communication, let alone forwarding any of the application data in unauthenticated form from an intermediate device to a server.

For at least these reasons, Jardin in view of Lockhart et al. and in further view of Lin fails to establish a prima facie case for non-patentability of Applicants' claims under 35 U.S.C. 103(a). Withdrawal of this rejection is requested.

## CONCLUSION

All claims in this application are in condition for allowance. Applicants respectfully request reconsideration and prompt allowance of all pending claims. Please charge any additional fees or credit any overpayment to deposit account number 50-1778. The Examiner is invited to telephone the below-signed attorney to discuss this application.

Date: May 23, 2006

By: Kent J. Sieffert

SHUMAKER & SIEFFERT, P.A.
8425 Seasons Parkway, Suite 105
St. Paul, Minnesota 55125
Telephone: 651.735.1100
Facsimile: 651.735.1102

Name: Kent J. Sieffert
Reg. No.: 41,312